CS 7545: Machine Learning Theory **Fall 2021**

Lecture 2

Name: Tyler LaBonte **Professor:** Santosh S. Vempala

1 Gaussian Mixture Models

Last lecture, we saw that it is simple to estimate a single Gaussian. Now, we consider what happens if each data point is drawn from one of k Gaussians at random. This simulates when the data is drawn from multiple distinct populations (in fact, the model was developed by Pearson for studying different species of crabs).

Formally, a k-GMM F comprises k Gaussians F_i with weights $w_i \geq 0$, where $F_i = \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ and $\sum_i w_i = 1$. To sample from a k-GMM, we sample from an F_i according to the probabilities $\boldsymbol{w}$.

To estimate a k-GMM, we would ideally like to estimate every $w_i, \boldsymbol{\mu}_i$, and $\boldsymbol{\Sigma}_i$. In fact, we will see that a mixture of Gaussians is uniquely identifiable, so this problem is well-defined. For now, we will focus on separable k-GMMs where we can partition the data according to its source component. Two notions of separability here are

1. Mean separation: $\|\mu_i - \mu_j\| > c \max\{\sigma_i, \sigma_j\}$.
2. Total variation distance: $d_{TV}(F_i, F_j) = \dfrac{1}{2}\displaystyle\int_{\mathbb{R}^d} |p_i(\boldsymbol{x}) - p_j(\boldsymbol{x})| d\boldsymbol{x}$.

Here, (1) implies (2) but not vice versa, as we could have two Gaussians with the same mean but vastly different variance. In fact, if the total variation distance is large, then either the mean separation is large or $\max\left\{\dfrac{\sigma_i}{\sigma_j}, \dfrac{\sigma_j}{\sigma_i}\right\}$ is large.

To simplify things, assume we are working with mean separation and $F_i = \mathcal{N}(\boldsymbol{\mu}, \sigma^2 \mathbf{I}_d)$. Then $\boldsymbol{x}, \boldsymbol{y} \sim F_i$ implies

$$\begin{aligned} \mathbb{E}[\|\boldsymbol{x} - \boldsymbol{y}\|^2] &= \mathbb{E}[\|\boldsymbol{x} - \boldsymbol{\mu}_i - (\boldsymbol{y} - \boldsymbol{\mu}_j)\|^2] && (1) \\ &= \mathbb{E}[\|\boldsymbol{x} - \boldsymbol{\mu}_i\|^2] + \mathbb{E}[\|\boldsymbol{y} - \boldsymbol{\mu}_j\|^2] && (2) \\ &= 2d\sigma^2 && (3) \end{aligned}$$

and $\boldsymbol{x} \sim F_i, \boldsymbol{y} \sim F_j$ implies

$$\begin{aligned} \mathbb{E}[\|\boldsymbol{x} - \boldsymbol{y}\|^2] &= \mathbb{E}[\|\boldsymbol{x} - \boldsymbol{\mu}_i - (\boldsymbol{y} - \boldsymbol{\mu}_j) + (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)\|^2] && (4) \\ &= 2d\sigma^2 + \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|^2. && (5) \end{aligned}$$

Thus, the mean separation allows us to differentiate whether a pair of points is from the same Gaussian. Our algorithm is then to put the closest pairs of points in the same cluster until there are k components. We'd like a high-probability guarantee on this algorithm.

Lemma 1.1: Gaussian Concentration

Suppose $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2 \mathbf{I}_d)$. Then for $t > 1$,

1. $\Pr[\|\boldsymbol{x} - \boldsymbol{\mu}\| > t\sqrt{d}\sigma] \leq 2e^{-t/2}$.
2. $\Pr[\|\boldsymbol{x} - \boldsymbol{\mu}\|^2 - d\sigma^2 > t\sqrt{d}\sigma^2] \leq 2e^{-t^2/8}$.

The first bound states that the density drops exponentially far from the mean, and the second bound states that the Gaussian is contained in a thin shell of constant width (even in high dimension).

Concentration tells us that with probability $1 - 2e^{-t^2/8}$, $\boldsymbol{x}, \boldsymbol{y} \sim F_i$ implies

$$\|\boldsymbol{x} - \boldsymbol{y}\|^2 \leq 2d\sigma^2 + t\sqrt{d}\sigma^2 \tag{6}$$

and $\boldsymbol{x} \sim F_i, \boldsymbol{y} \sim F_j$ implies

$$\|\boldsymbol{x} - \boldsymbol{y}\|^2 > 2d\sigma^2 + \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|^2 - t\sqrt{d}\sigma^2. \tag{7}$$

Thus, it suffices to have $\|\boldsymbol{\mu}_i - \boldsymbol{\mu} j\|^2 \geq 2t\sqrt{d}\sigma^2$.

By setting $t = \sqrt{c \log \frac{m}{\delta}}$ and taking a union bound over $\binom{m}{2}$ pairs of samples, we obtain the following theorem.

Theorem 1.1: Clustering Mean-Separable Gaussian Mixtures

For $\delta > 0$, with probability $1 - \delta$ a random sample of m points from a k-GMM with mean separation

$$\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\| > c\Big(\big(\log \frac{m}{\delta}\big)d\Big)^{1/4}\sigma \tag{8}$$

can be clustered correctly in polynomial time.

2 Improving the Bound

This is a good start but ultimately unsatisfactory. We'd like to remove the $d^{1/4}$ term since information-theoretically there should be no dependence on d; if we knew the line joining $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$, we would only need $c\sigma$ separation between them.

Suppose $\mathbf{A}$ is the matrix whose rows are the samples of a 2-GMM. Then, the line joining the means is

$$\begin{aligned} \boldsymbol{v}_1 &= \operatorname*{argmax}_{\|\boldsymbol{v}\|=1} \|\mathbf{A}\boldsymbol{v}\|^2 && (9) \\ &= \operatorname*{argmin}_{\boldsymbol{v}} \|\mathbf{A} - (\mathbf{A}\boldsymbol{v})\boldsymbol{v}^\intercal\|^2. && (10) \end{aligned}$$

Definition 2.1: Singular Vector

$(\boldsymbol{u}, \boldsymbol{v})$ are a left/right singular vector pair of $\mathbf{A}$ if $\mathbf{A}\boldsymbol{v} = \sigma\boldsymbol{u}$ and $\mathbf{A}^\intercal\boldsymbol{u} = \sigma\boldsymbol{v}$ for $\sigma > 0$. While eigenvectors describe the invariant action of $\mathbf{A}$, singular vectors describe the maximal action of $\mathbf{A}$. Here, σ is called a singular value.

Lemma 2.1: Best-fit and Singular Vectors

$\boldsymbol{v}_1$ is the right singular vector of $\mathbf{A}$ with maximal singular value.

Proof: Since $\mathbf{A}^\intercal\mathbf{A}\boldsymbol{v} = \sigma\mathbf{A}^\intercal\boldsymbol{u} = \sigma^2\boldsymbol{v}$, we know that $\boldsymbol{v}$ is the eigenvector of $\mathbf{A}^\intercal\mathbf{A}$ with maximal eigenvalue σ^2. This goes both ways by defining $\boldsymbol{u} = \frac{1}{\sigma}\mathbf{A}\boldsymbol{v}$. Suppose $f(\boldsymbol{v}) = \|\mathbf{A}\boldsymbol{v}\|^2 = \boldsymbol{v}\mathbf{A}^\intercal\mathbf{A}\boldsymbol{v}$, then $\nabla_{\boldsymbol{v}} f(\boldsymbol{v}) = \lambda\mathbf{A}^\intercal\mathbf{A}\boldsymbol{v} = \lambda\boldsymbol{v}$. So all extrema are eigenvectors, and thus the maximizer is the largest. ■